

**ІНФАРМАТЫКА**

УДК 519.714

*Л. Д. ЧЕРЕМИСИНОВА***ПОИСК КРАТЧАЙШЕЙ УСТАНОВОЧНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ СХЕМЫ  
С ПАМЯТЬЮ НА D-ТРИГГЕРАХ**

*Объединенный институт проблем информатики Национальной академии наук Беларуси, Минск, Беларусь,  
e-mail: cld@newman.bas-net.by*

Рассматривается задача поиска установочной последовательности наименьшей длины для логической схемы с памятью на D-триггерах. Предлагается метод сведения этой проблемы к задаче булевой выполнимости, которая может быть эффективно решена с помощью SAT-решателей. Метод основан на построении конъюнктивной нормальной формы разрешения комбинационного блока, реализующего функции возбуждения триггеров.

*Ключевые слова:* тестирование, верификация, логический синтез, автоматизация проектирования.

*L. D. CHEREMISINOVA***SHORTEST SYNCHRONIZING SEQUENCE SEARCH FOR A SEQUENTIAL NETWORK  
WITH MEMORY ON D FLIP-FLOPS**

*United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk, Belarus,  
e-mail: cld@newman.bas-net.by*

The problem under consideration is to find a synchronizing sequence of a minimal size for a logical network having flip-flop primitives of type D as memory elements. A novel method is proposed, which is based on the formulation of the task as the Boolean satisfiability problem solved with any standard SAT-solver. The method is based on forming the conventional conjunctive normal form representation for combinational block, implementing excitation functions of the flip-flops.

*Keywords:* testing, verification, logic synthesis, design automation.

**Введение.** Успехи в области микроэлектроники привели к резкому возрастанию сложности проектируемых сверхбольших интегральных схем (СБИС), что значительно усложнило задачу обеспечения правильности их функционирования. По мере усложнения СБИС все более ответственным этапом проектирования становится их диагностирование, обеспечивающее оперативное обнаружение ошибок в устройствах в процессе их работы. Значительная часть времени проектирования цифровой аппаратуры уходит на разработку проверяющих и диагностических тестов.

Традиционный метод анализа цифровой аппаратуры основан на моделировании поведения на последовательности входных воздействий (тестовой последовательности) и сравнении полученных откликов на выходных полюсах устройства с ожидаемыми. Относительно просто выполняется тестирование (и строится тест) для случая комбинационных схем. Наличие элементов памяти у последовательностных схем значительно усложняет их тестирование по сравнению с комбинационными в силу того, что поведение последовательностных схем в значительной степени зависит от их внутреннего состояния (набора состояний элементов памяти), в котором они находятся перед подачей на входы тестовых воздействий.

В промышленности используется множество методов и программно-аппаратных средств автоматизации процесса тестирования, включающих анализ тестируемости устройств, автоматическую

генерацию тестовых наборов (ATPG – automatic test pattern generation), моделирование неисправностей, проектирование тестируемых схем (DFT – design for testability), встроенное самотестирование (BIST – built-in self-test) [1–5]. В настоящее время наиболее перспективным направлением считается развитие методов и средств встроенного самотестирования [1, 3]. Эти методы основаны на использовании встроенных генераторов тестовых последовательностей, получении сигнатуры реакций тестируемой схемы путем сжатия этих последовательностей и сравнении полученной и эталонной сигнатур. Достоинство этого вида тестирования заключается в высокой скорости и сокращении объема сравниваемой информации. Недостатком же является необходимость встраивания дополнительной аппаратуры и возможный пропуск тестовых наборов, на которых проявляется неисправность, за счет их возможного попадания в обратную (исправную) сигнатуру и неполного охвата пространства возможных входных наборов [3, 4]. Это становится проблемой при тестировании устройств, для которых требуется большое покрытие тестами (99 % или выше) [3]. В таких случаях наиболее приемлема технология внешнего тестирования на основе направленной генерации наборов тестовой последовательности.

Перед началом тестирования необходимо провести инициализацию тестируемой последовательностной схемы, заключающуюся в установлении элементов памяти в некоторые известные состояния путем подачи на входы схемы некоторой последовательности воздействий, которая называется установочной [4–6]. Под установочной последовательностью далее понимается последовательность входных воздействий, переводящая схему в некоторое известное определенное состояние (при этом получаемая выходная последовательность может быть любой, так как она зависит от начального состояния схемы).

Обычно задача построения установочных последовательностей рассматривается для случая, когда поведение последовательностной схемы задается в виде дискретного автомата [7, 8]. Обзор существующих методов построения установочных последовательностей для конечных автоматов можно найти в работе [9]. Следует отметить, что поиск кратчайшей установочной последовательности (минимальной длины) представляет собой NP-трудную задачу [10].

В настоящей статье задача построения кратчайшей установочной последовательности рассматривается в более общем виде – для структурного описания последовательностной схемы. Под состоянием схемы понимается набор состояний, в которых находятся ее элементы памяти. Предполагается, что начальное состояние элементов памяти схемы неизвестно. Необходимо найти такую последовательность наборов значений сигналов, подаваемых на входы схемы в следующие друг за другом такты, которая переводит схему в известное состояние и имеет минимальную длину. В работе рассматривается случай, когда схема является синхронной, элементы памяти – D-триггеры, а комбинационные элементы – как простые вентили типа И-НЕ, ИЛИ-НЕ, И, ИЛИ, так и элементы из некоторой библиотеки, реализующие более сложные булевы функции. Следует заметить, что решение этой задачи не всегда существует, но иногда она может быть упрощена: необходимо найти последовательность, устанавливающую один триггер в заданное состояние (0 или 1).

Предлагаемый в данной работе метод основан на сведении задачи поиска установочной последовательности к задаче выполнимости конъюнктивной нормальной формы (КНФ), которая может быть эффективно решена с помощью известных SAT-решателей [11–13], обеспечивающих проверку выполнимости КНФ значительной сложности.

**1. Постановка задачи.** Последовательностную схему часто представляют в виде двухблочной структуры: блока комбинационной логики и блока триггеров. Первый блок является чисто комбинационной схемой, часть его выходов служит входами блока триггеров. Выходы триггеров, в свою очередь, являются входами комбинационного блока схемы (рис. 1) в следующем такте работы схемы. Таким образом, комбинационный блок имеет два типа входных полюсов: внешние входы последовательностной схемы, значения которых определяются сигналами извне, и внутренние полюсы, значения которых определяются сигналами с выходных полюсов триггеров и задают состояние элементов памяти. Аналогично комбинационный блок имеет и два типа выходных полюсов: внешние выходы последовательностной схемы, значения которых передаются во внешнее окружение последовательностной схемы, и внутренние выходы, значения которых передаются на блок памяти и определяют функции возбуждения триггеров.

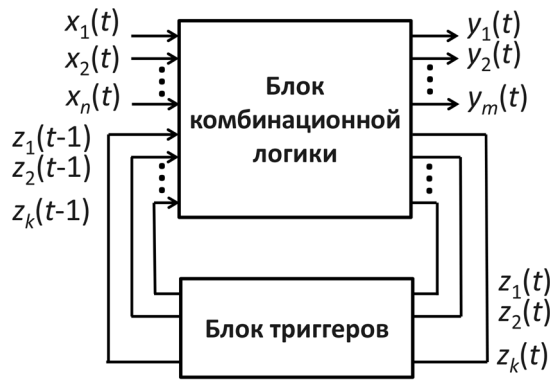


Рис. 1. Структура последовательной схемы

Будем считать далее, что комбинационный блок состоит из элементов, таких как И, ИЛИ, НЕ, И-НЕ, ИЛИ-НЕ, исключающее ИЛИ, либо из более сложных элементов некоторой библиотеки, блок памяти состоит из D-триггеров. Пример такой схемы показан на рис. 2. Задача заключается в инициализации блока памяти, т. е. в поиске установочной последовательности, подача которой на входы схемы гарантированно устанавливает все элементы памяти в известные состояния.

Пусть имеем синхронную схему с  $n$  входными полюсами и  $k$  D-триггерами. Обозначим через  $\mathbf{x}^i = (x_1^i, x_2^i, \dots, x_n^i)$  булев вектор значений входных сигналов, подаваемых на входы схемы в момент времени  $i$ . Последовательность входных воздействий  $\mathbf{X} = (\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^p)$  называется установочной, если состояние (определяемое состоянием всех триггеров), в которое переходит схема после подачи на ее входы этой последовательности, определяется однозначно вне зависимости от того, в каком состоянии схема была перед началом эксперимента. Число  $p$  называется длиной установочной последовательности. Установочная последовательность наименьшей длины называется кратчайшей.

Задача состоит в поиске кратчайшей установочной последовательности для синхронной последовательностной схемы выделенного типа. Следует заметить, что для любой заданной схемы может существовать не одна, а несколько установочных последовательностей (в том числе и кратчайших).

Поведение D-триггера описывается автоматом Мура, который имеет два состояния: обозначаемые через  $Q$  и  $\bar{Q}$  и соответствующие 1 и 0. Значение сигнала на выходе D-триггера совпадает с состоянием, в котором он находится в соответствующий момент времени. D-триггер выполняет роль элемента задержки сигнала. Если значение функции возбуждения входа триггера в момент времени  $t$  равно  $z$ , то в следующий момент времени  $t+1$  он переходит в состояние  $z$  и на его выходе появляется сигнал  $z$ . Соответственно поиск установочной последовательности, переводящей D-триггеры в заданные состояния, сводится к поиску входной последовательности, обеспечивающей соответствующие значения функций их возбуждения.

Таким образом, можно ограничиться рассмотрением только комбинационной части схемы, которая ответственна за генерацию функций возбуждения триггеров (рис. 3). Выделенная и рассматриваемая далее схема имеет  $n + k$  входов, описываемых переменными  $x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_k$

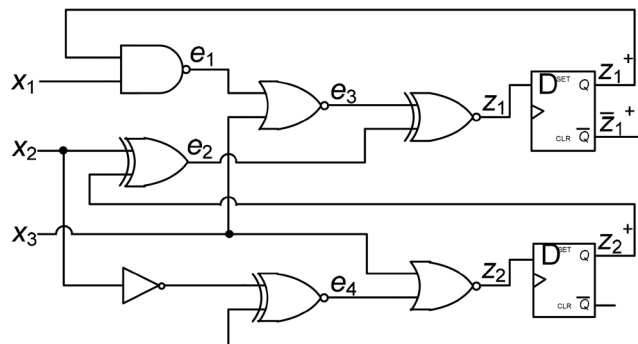


Рис. 2. Пример тестовой последовательностной схемы

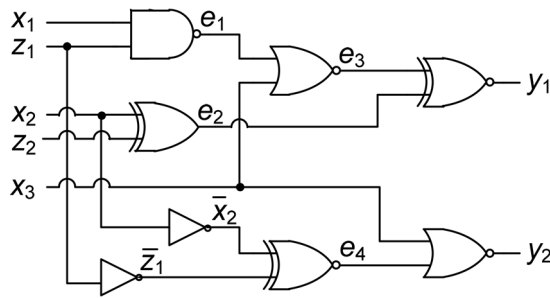


Рис. 3. Комбинационная часть тестовой последовательностной схемы

и соответствующих  $n$  входным полюсам схемы и  $k$  выходным полюсам триггеров, и  $k$  выходов, описываемых переменными  $y_1, y_2, \dots, y_k$  и определяющих  $k$  функций возбуждения триггеров. Значения выходных переменных  $y_1, y_2, \dots, y_k$  рассматриваемой комбинационной схемы будут являться значениями ее входных переменных  $z_1, z_2, \dots, z_k$  на следующем такте работы.

**2. КНФ разрешения логической схемы.** КНФ представляет булеву функцию в виде конъюнкции одного или более дизъюнктов. Каждый дизъюнкт, в свою очередь, является дизъюнкцией одного или более литералов (под литералом понимается булева переменная или ее отрицание). КНФ задает полностью определенную булеву функцию, и каждый из ее дизъюнктов соответствует имплиценте этой функции. Матричное представление КНФ задается троичной матрицей, строки которой соответствуют дизъюнктам, а столбцы – переменным.

Задача проверки выполнимости КНФ заключается в нахождении такого присваивания (может быть частичным) значений ее переменным, которое обращает КНФ в 1. Если такое присваивание удастся найти, то говорят, что КНФ выполнима и полученное присваивание называют выполняющим эту КНФ. Иначе КНФ невыполнима и представляет собой функцию, тождественно равную 0.

Множество равенств типа  $x_i = \sigma_i$  (где  $\sigma_i \in \{0, 1\}$ ,  $i = \{1, 2, \dots, n\}$ ) задает присваивание значений компонентам вектора  $x = (x_1, x_2, \dots, x_n)$ . Присваивание значений компонентам вектора  $x$  может быть полным, если задаются значения всех  $x_i$ , или частичным в противном случае. Полное присваивание представляет собой элемент (набор), а частичное – интервал  $n$ -мерного булева пространства  $E^n$ . Интервал ранга  $p$  фиксирует значения  $p$  переменных и покрывает  $2^{n-p}$  элементов булева пространства, также его можно представить в виде конъюнкции  $p$  литералов.

Структура комбинационной схемы задается направленным ациклическим графом, в котором вершины соответствуют логическим элементам, внешним входам и выходам схемы, а ребра – соединениям элементов и внешних полюсов. Большинство SAT-решателей исходят из представления схемы в виде так называемой КНФ разрешения схемы [14], которая задает все возможные допустимые комбинации сигналов на всех ее  $n + l$  полюсах, где  $n$  – число входных полюсов схемы,  $l$  – число элементов. При построении КНФ разрешения схемы выходные полюсы элементов и сами элементы обозначаются одним и тем же именем. Локальная функция элемента (полюса схемы)  $y$  определяет зависимость сигнала на выходе элемента от сигналов на его входах (т. е. задается в терминах локальных входных переменных). Функцией разрешения элемента, реализующего локальную функцию  $y = f(z_1, z_2, \dots, z_n)$ , называется функция  $\varphi(y, f) = y \sim f(z_1, z_2, \dots, z_n)$ , задающая все допустимые комбинации сигналов на полюсах этого элемента. Представление функции  $\varphi(y, f)$  в форме КНФ дает КНФ разрешения элемента. Ниже приводятся КНФ разрешения некоторых простых элементов:

$$\begin{aligned}
 y = \bar{z} &\rightarrow (z \vee y) (\bar{z} \vee \bar{y}); \\
 y = z_1 \oplus z_2 &\rightarrow (z_1 \vee z_2 \vee \bar{y}) (\bar{z}_1 \vee \bar{z}_2 \vee \bar{y}) (z_1 \vee \bar{z}_2 \vee y) (\bar{z}_1 \vee z_2 \vee y); \\
 y = z_1 z_2 \dots z_n &\rightarrow (z_1 \vee \bar{y}) (z_2 \vee \bar{y}) \dots (z_n \vee \bar{y}) (\bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n \vee y); \\
 y = \bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n &\rightarrow (\bar{z}_1 \vee y) (\bar{z}_2 \vee y) \dots (\bar{z}_n \vee y) (z_1 \vee z_2 \vee \dots \vee z_n \vee \bar{y}); \\
 y = \bar{z}_1 \oplus \bar{z}_2 &\rightarrow (z_1 \vee z_2 \vee y) (\bar{z}_1 \vee \bar{z}_2 \vee y) (z_1 \vee \bar{z}_2 \vee \bar{y}) (\bar{z}_1 \vee z_2 \vee \bar{y}); \\
 y = \bar{z}_1 z_2 \dots z_n &\rightarrow (z_1 \vee y) (z_2 \vee y) \dots (z_n \vee y) (\bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n \vee \bar{y}); \\
 y = \bar{z}_1 \vee \bar{z}_2 \vee \dots \vee \bar{z}_n &\rightarrow (\bar{z}_1 \vee \bar{y}) (\bar{z}_2 \vee \bar{y}) \dots (\bar{z}_n \vee \bar{y}) (z_1 \vee z_2 \vee \dots \vee z_n \vee y).
 \end{aligned}$$

КНФ разрешения всех элементов схемы объединяются операцией конъюнкции в одну КНФ разрешения схемы. Ее размер и сложность данного преобразования линейно зависят от числа логических элементов комбинационной схемы. КНФ разрешения  $(z \vee y)$  ( $\overline{z \vee y}$ ) инвертора  $y = \overline{z}$  можно опускать, заменяя все вхождения переменной  $y$  в КНФ разрешения схемы на  $\overline{z}$ . В первом столбце табл. 1 приведена КНФ  $C^1$  разрешения тестовой схемы (см. рис. 3).

Таблица 1. КНФ разрешения комбинационной схемы

№	КНФ $C^1$												КНФ $C^1_{пр}$ после упрощения $C^1$											
	$z_1^1$	$z_2^1$	$x_1^1$	$x_2^1$	$x_3^1$	$e_1^1$	$e_2^1$	$e_3^1$	$e_4^1$	$y_1^1$	$y_2^1$	$z_1^1$	$z_2^1$	$x_1^1$	$x_2^1$	$x_3^1$	$e_1^1$	$e_2^1$	$e_3^1$	$e_4^1$	$y_1^1$	$y_2^1$		
	1												2											
1	-	-	1	-	-	1	-	-	-	-	-	$e_1^1$	-	-	-	-	-	1	-	-	-	-	-	$e_1^1$
2	1	-	-	-	-	1	-	-	-	-	-		-	-	0	-	-	-	-	-	-	-	-	
3	0	-	0	-	-	0	-	-	-	-	-		-	-	-	-	-	0	-	0	-	-	-	$e_3^1$
4	-	1	-	1	-	-	1	-	-	-	-	$e_2^1$	-	-	-	-	0	-	-	0	-	-	-	
5	-	0	-	0	-	-	1	-	-	-	-		-	-	-	-	1	1	-	1	-	-	-	
6	-	1	-	0	-	-	0	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-	$y_1^1$
7	-	0	-	1	-	-	0	-	-	-	-		-	-	-	-	-	-	-	-	-	-	0	$y_2^1$
8	-	-	-	-	-	0	-	0	-	-	-	$e_3^1$	-	-	-	-	1	-	-	-	-	-	-	
9	-	-	-	-	0	-	-	0	-	-	-													
10	-	-	-	-	1	1	-	1	-	-	-													
11	0	-	-	0	-	-	-	-	1	-	-	$e_4^1$												
12	1	-	-	1	-	-	-	-	1	-	-													
13	0	-	-	1	-	-	-	-	0	-	-													
14	1	-	-	0	-	-	-	-	0	-	-													
15	-	-	-	-	-	-	1	1	-	1	-	$y_1^1$												
16	-	-	-	-	-	-	0	0	-	1	-													
17	-	-	-	-	-	-	1	0	-	0	-													
18	-	-	-	-	-	-	0	1	-	0	-													
19	-	-	-	-	0	-	-	-	-	-	0	$y_2^1$												
20	-	-	-	-	-	-	-	-	0	-	0													
21	-	-	-	-	1	-	-	-	1	-	1													

**3. Метод поиска установочной последовательности.** Задача булевой выполнимости формулируется относительно двоичных переменных, которые представляют собой ограничения, накладываемые решаемой проблемой. Все присваивания значений переменным, выполняющие КНФ, эквивалентны с точки зрения решения задачи выполнимости. Соответственно в процессе поиска установочной последовательности посредством решения задачи выполнимости КНФ не существует возможности сформулировать эту задачу таким образом, чтобы можно было найти кратчайшую установочную последовательность или сказать, что ее для данной схемы не существует. Можно только сформулировать задачу поиска установочной последовательности некоторой априори заданной длины в виде задачи о выполнимости некоторой КНФ, получить один (или все) набор значений булевых переменных, выполняющий эту КНФ, если КНФ выполнима. Значения входных переменных схемы в выполняющем наборе зададут установочную последовательность. Если КНФ окажется невыполнимой, то установочной последовательности заданной длины не существует.

Таким образом, задача поиска кратчайшей установочной последовательности может быть сформулирована как ряд задач поиска установочных последовательностей возрастающей длины. Процесс заканчивается, когда на некотором шаге будет установлена выполнимость анализируемой КНФ и, соответственно, найдена кратчайшая установочная последовательность. Ясно, что априори должна быть задана и предельная длина искомой последовательности. Далее будет предложен метод, позволяющий сократить вычислительные затраты при поиске кратчайшей установочной

последовательности путем сведения к задаче проверки выполнимости КНФ. Покажем, как построить первую КНФ разрешения  $C^1$  комбинационной части схемы при проверке существования установочной последовательности длины 1, а затем приведем метод итеративной модификации КНФ разрешения  $C^i$  для получения КНФ  $C^{i+1}$  при необходимости увеличения длины искомой установочной последовательности.

Процесс поиска кратчайшей установочной последовательности начинается с поиска установочной последовательности  $X^1 = (x^1)$  длины 1, т. е. последовательности входных воздействий, приводящей к установке триггеров в известное состояние за один такт работы схемы. При построении КНФ разрешения  $C^1$  для комбинационной части схемы предполагается, что аргументы  $(x_1^1, x_2^1, \dots, x_n^1)$  соответствуют входным полюсам последовательностной схемы, аргументы  $(z_1^1, z_2^1, \dots, z_k^1)$  соответствуют  $k$  линиям обратной связи последовательностной схемы (с выходных полюсов триггеров). Предполагается, что значения входных переменных  $(z_1^1, z_2^1, \dots, z_k^1)$  комбинационной части схемы перед началом эксперимента по поиску установочной последовательности не определены, т. е.:  $z_1^1 = z_2^1 = \dots = z_k^1 = \langle\langle - \rangle\rangle$ , так как начальное состояние последовательностной схемы неизвестно.

Если построенная таким образом КНФ разрешения  $C^1$  выполнима, то будет найден выполняющий набор значений всех ее переменных, в частности набор значений переменных  $X^1 = (x_1^1, x_2^1, \dots, x_n^1)$ , представляющий кратчайшую установочную последовательность (длины 1), и набор значений переменных  $Y^1 = (y_1^1, y_2^1, \dots, y_k^1)$ , задающий значения выходных функций комбинационного блока или значения функций возбуждения триггеров в ответ на подачу на входы значений из  $X^1$ . Значения функций возбуждения определяют состояние последовательностной схемы в следующий такт ее работы.

Если построенная КНФ разрешения  $C^1$  невыполнима, то длина искомой установочной последовательности увеличивается на 1 и формируется КНФ разрешения  $C^2$  для поиска установочной последовательности  $X^2 = (x^1, x^2)$ ,  $Y^2 = (y^1, y^2)$ . При этом КНФ  $C^2$  формируется для двухблочной комбинационной схемы (первых два блока на рис. 4). Внутренняя структура обоих блоков – это одна и та же комбинационная часть исходной последовательностной схемы (см. рис. 3). Входными переменными второго блока являются переменные  $x_1^2, x_2^2, \dots, x_n^2$ , подаваемые на входы последовательностной схемы на втором такте ее работы, и переменные  $z_1^2, z_2^2, \dots, z_k^2$ , соответствующие линиям обратной связи, их значения совпадают на втором такте со значениями выходных переменных первого блока:  $z_1^2 = y_1^1, z_2^2 = y_2^1, \dots, z_k^2 = y_k^1$ . Если полученная КНФ разрешения  $C^2$  выполнима, то будет найдена установочная последовательность  $X^2 = (x^1, x^2)$  длины 2, выполняющая установку триггеров в известные состояния (определяемые набором  $Y^2 = (y^1, y^2)$ ) за два такта.

Если КНФ  $C^2$  невыполнима, то аналогичным образом производится увеличение числа блоков анализируемой комбинационной схемы (см. рис. 4) и соответственно ищется установочная последовательность  $X^3 = (x^1, x^2, x^3)$  длины 3. Процесс увеличения числа блоков комбинационной схемы продолжается до тех пор, пока на некотором шаге не будет найдена установочная последовательность, которая и будет кратчайшей, или число итераций не достигнет наперед заданного предельного числа, и тогда установочная последовательность не будет найдена.

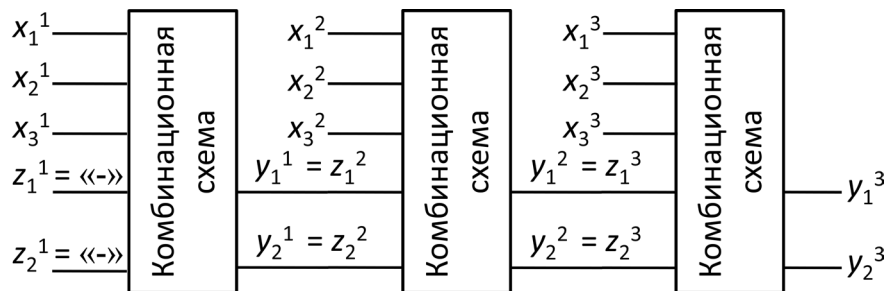


Рис. 4. Увеличение числа блоков комбинационной логики до трех

#### 4. Поиск установочной последовательности посредством сведения к выполнимости КНФ.

Так как переменные  $z_1^i, z_2^i, \dots, z_k^i$  КНФ  $C^i$  отождествляются с переменными  $y_1^{i-1}, y_2^{i-1}, \dots, y_k^{i-1}$ , то для упрощения процесса построения  $(i+1)$ -й КНФ по  $i$ -й (в случае невыполнимости  $C^i$ ) примем следующий порядок столбцов матричного задания КНФ:  $z_1^i, z_2^i, \dots, z_k^i, x_1^i, x_2^i, \dots, x_n^i, e_1^i, e_2^i, \dots, e_l^i, y_1^i, y_2^i, \dots, y_k^i$ .

Рассмотрим для примера приведенную выше последовательностную схему (см. рис. 2), для которой найдем кратчайшую установочную последовательность. Согласно изложенному выше методу, построим сначала одноблочную схему (см. рис. 3), приписав всем символам переменных верхний индекс 1. КНФ разрешения  $C^1$  этой схемы определена на 11 аргументах и состоит из 21 дизъюнкта (первый столбец табл. 1).

Перед проверкой выполнимости необходимо заменить определенные значения переменных  $z_1^1, z_2^1, \dots, z_k^1$  на неопределенные, так как их значения перед началом эксперимента неизвестны и в выполняющем КНФ наборе им нельзя присвоить определенные значения в отличие от переменных  $x_1^1, x_2^1, x_3^1$ . Замена значений этих переменных упрощает КНФ  $C^1$ : некоторые строки становятся избыточными. Например, три первых дизъюнкта  $(x_1^1 \vee e_1^1)(z_1^1 \vee e_1^1)(\bar{z}_1^1 \vee x_1^1 \vee \bar{e}_1^1)$ , порождаемых КНФ разрешения элемента  $e_1^1$ , заменяются сначала на три более простых дизъюнкта  $(x_1^1 \vee e_1^1)e_1^1(\bar{x}_1^1 \vee \bar{e}_1^1)$ , а затем после упрощения – на два:  $e_1^1 \bar{x}_1^1$ .

Следует заметить, что существуют простейшие элементы, сигналы на выходных полюсах которых имеют неопределенное значение в том случае, если хотя бы один входной сигнал элемента имеет неопределенное значение. Таким элементом является исключающее ИЛИ. Например, четыре дизъюнкта  $(z_2^1 \vee x_2^1 \vee e_2^1)(\bar{z}_2^1 \vee \bar{x}_2^1 \vee e_2^1)(z_2^1 \vee \bar{x}_2^1 \vee \bar{e}_1^1)(\bar{z}_2^1 \vee x_2^1 \vee \bar{e}_1^1)$  для КНФ разрешения элемента  $e_2^1$  после удаления переменных  $z_2^1$  заменяются на  $(x_2^1 \vee e_2^1)(\bar{x}_2^1 \vee e_2^1)(\bar{x}_2^1 \vee \bar{e}_1^1)(x_2^1 \vee \bar{e}_1^1)$ . Эта последняя КНФ невыполнима. Следовательно, можно удалить эти дизъюнкты из КНФ  $C^1$ , распространив неопределенное значение переменной  $e_2^1$  по схеме путем исключения всех вхождений этой переменной в дизъюнкты КНФ  $C^1$ . Это также упростит вид КНФ  $C^1$ . Упрощенное представление  $C^1_{пр}$  КНФ разрешения  $C^1$  для одноблочной схемы (см. рис. 3) показано во втором столбце табл. 1.

В нашем случае КНФ  $C^1_{пр}$  оказывается невыполнимой. Это означает, что установочной последовательности длины 1 для последовательностной схемы (см. рис. 2) не существует. Если точнее, то в рассматриваемом случае не удалось инициализировать первый D-триггер за один такт, но существует установочная последовательность (в данном случае набор значений входных переменных схемы) длины 1 для второго D-триггера: при  $x_3^1 = 1, y_2^1 = 0$ . Это следует из того, что КНФ разрешения  $C^1_{пр}$  для одноблочной схемы (см. рис. 3), реализующей функцию  $y_2^1$  возбуждения второго триггера, состоит из двух дизъюнкта:  $y_2^1 x_3^1$  (как это следует из табл. 1).

Для поиска установочной последовательности длины 2 строим двухблочную схему (рис. 5), состоящую из пары одинаковых последовательно связанных блоков (см. рис. 3), две пары полюсов для которых отождествлены:  $y_1^1$  и  $z_1^2, y_2^1$  и  $z_2^2$ . КНФ разрешения  $C^2$  (первый столбец табл. 2)

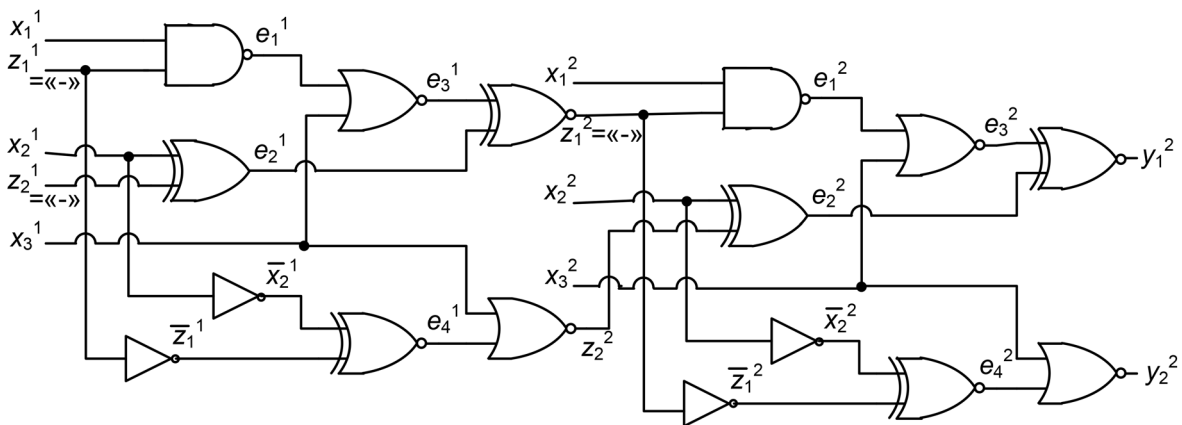


Рис. 5. Двухблочная комбинационная схема для поиска установочной последовательности длины 2 для последовательностной схемы





Для того чтобы инициализировать последовательностную схему (см. рис. 2), необходимо на первом такте подать на ее входы сигналы  $x_1 = 0, x_2 = 0, x_3 = 1$  или  $x_1 = 0, x_2 = 1, x_3 = 1$ , а на втором такте –  $x_1 = 0, x_2 = 0, x_3 = 1$ . В результате оба D-триггера схемы установятся в состояние 0.

**5. Особенности поиска установочной последовательности посредством сведения к выполнимости КНФ.** Следует обратить внимание на некоторые особенности поиска установочной последовательности посредством булевой выполнимости. Они обусловлены наличием обратных связей в последовательностной схеме, которые приводят к неопределенности значений тех входных сигналов комбинационной части схемы, которые снимаются с выходов триггеров на следующем такте работы. На первом такте инициализации последовательностной схемы соответствующим переменным не могут быть произвольно присвоены никакие определенные значения, так как их значения определены в процессе предшествующего эксперименту функционирования схемы и неизвестны. Исходя из этого приходится исключать переменные  $z_1^1, z_2^1, \dots, z_k^1$  из КНФ разрешения  $C^1$  одноблочной комбинационной схемы. Исключение этих переменных может привести к невыполнимости некоторых фрагментов КНФ  $C^1$  в силу того, что неопределенность значений сигналов на одном или нескольких входных полюсах некоторых элементов может привести к неопределенности сигнала на их выходных полюсах. Характерным примером служит КНФ разрешения элемента типа исключающее ИЛИ. Неопределенность на одном из его входов приводит к появлению неопределенности на его выходе и, соответственно, к невыполнимости КНФ разрешения этого элемента. При построении КНФ разрешения элемента типа исключающее ИЛИ можно сразу не включать его КНФ разрешения в КНФ разрешения  $C^1$  одноблочной комбинационной схемы, перенося неопределенность на выход элемента и присваивая неопределенное значение соответствующей переменной во всех дизъюнктах КНФ  $C^1$ .

Для того чтобы решить означенную проблему в общем случае, можно воспользоваться для анализа выполнимости КНФ разрешения  $C^i$  SAT-решателем (например, PicoSAT [11]), который после установления конфликта в процессе анализа на выполнимость выдает путь доказательства, приведший к его выявлению. Получив фрагмент, являющийся причиной невыполнимости КНФ  $C^i$ , можно исключить его из КНФ  $C^i$ , присвоив неопределенное значение переменной, соответствующей выходному полюсу фрагмента во всех остальных дизъюнктах КНФ.

**Заключение.** Рассмотрена задача поиска кратчайшей установочной последовательности для синхронной последовательностной схемы с памятью на D-триггерах. Предложен метод сведения этой проблемы к задаче булевой выполнимости КНФ разрешения комбинационной схемы, реализующей функции возбуждения D-триггеров. Метод основан на последовательном увеличении числа однотипных блоков комбинационной схемы, в которой  $i$ -й блок соответствует  $i$ -у такту работы последовательностной схемы. Процесс увеличения числа блоков комбинационной схемы продолжается до тех пор, пока на некотором шаге не будет получена выполнимая КНФ разрешения, по которой и будет определена кратчайшая установочная последовательность.

### Список использованной литературы

1. *Strunz, B.* Design for Testability in Digital Integrated circuits [Electronic resource] / B. Strunz, C. Flanagan, T. Hall; University of Limerick, Ireland. – Mode of access: [http://www.cs.colostate.edu/~cs530/digital\\_testing.pdf](http://www.cs.colostate.edu/~cs530/digital_testing.pdf). – Date of access: 01.07.2015.
2. Logic BIST: State-of-the-Art and Open Problems [Electronic resource] / N. Li [et al.]. – Mode of access: <http://arxiv.org/pdf/1503.04628.pdf>. – Date of access: 01.07.2015.
3. *Chakrabarty, K.* DFBT: A Design-for-Testability Method Based on Balance Testing [Electronic resource] / K. Chakrabarty, P. Hayes; Department of Electrical Engineering and Computer Science University of Michigan. – Mode of access: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.380.9873&rep=rep1&type=pdf>. – Date of access: 01.07.2015.
4. *Rene, D.* Random testing of digital circuits. Theory and application / D. Rene. – [S. l.]: Marcel Dekker, Inc. 1998.
5. *Crouch, A.* Design-for-Test for Digital IC's and Embedded Core Systems / A. Crouch. – [S. l.]: Prentice Hall, 1999.
6. *Bushnell, M. L.* Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits / M. L. Bushnell, V. D. Agrawal. – [S. l.]: Kluwer Academic Publishers, 2002.
7. *Kohavi, Z.* Switching and Finite Automata Theory / Z. Kohavi. – 2nd ed. – Cambridge [et al.]: Cambridge Univ. Press, 1978.
8. *Gill, A.* Introduction to the Theory of Finite-state Machines / A. Gill. – [S. l.]: McGraw-Hill, 1962.

9. *Lee, D.* Principles and methods of testing finite state machine – a survey / D. Lee, M. Yannakakis // Proc. of the IEEE. – 1996. – Vol. 84 (8). – P. 1090–1123.
10. *Eppstein, D.* Reset sequences for monotonic automata / D. Eppstein // SIAM J. on Computing. – 1990. – Vol. 19, no. 3. – P. 500–510.
11. *Biere, A.* PicoSAT essentials / A. Biere // J. on Satisfiability, Boolean Modeling and Computation. – 2008. – Vol. 4. – P. 75–97.
12. *Mahajan, Y.* Zchaff 2004: an efficient SAT solver / Y. Mahajan, Z. Fu, S. Malik // Theory and Applications of Satisfiability Testing (2004 SAT Solver Competition and QBF Solver Evaluation (Invited Papers)). – Berlin; Heidelberg: Springer, 2005. – P. 360–375.
13. *Goldberg, E.* BerkMin: a fast and robust SAT-solver / E. Goldberg, Y. Novikov // 2002 Design, Automation and Test in Europe Conference and Exposition (DATE 2002), 4–8 March 2002, Paris, France. – [S. l.]: IEEE Computer Society, 2002. – P. 142–149.
14. *Kuehlmann, A.* Combinational and Sequential Equivalence Checking / A. Kuehlmann A., C. A. J. van Eijk. // Logic synthesis and Verification / eds S. Hassoun, T. Sasao, R. K. Brayton. – [S. l.]: Kluwer Academic Publishers, 2002. – P. 343–372.

*Поступила в редакцию 25.08.2015*